

RCA IN IT: ROOT CAUSE ANALYSIS FOR IT ENVIRONMENTS



In the world of technology and software development, you are always trying out something new—only to test it again. [Engineers](#) learn from their mistakes and use them to grow their skillsets and improve processes. But some mistakes, like a major network or infrastructure failure, are less forgiving. The result of these unintended problems is a thing of nightmares.

Fortunately, a systematic approach available helps engineers and developers find the beginning of a problem and discover what went wrong: root cause analysis. In this article, we'll look at RCA in IT environments, including:

- [Defining RCA and why it might be necessary](#)
- [Exploring RCA strategies, including the 5 Whys](#)
- [Understanding the many benefits of RCA](#)

What is root cause analysis?

Root cause analysis (RCA) is a systematic process for finding and identifying the root cause of a problem or event.

RCA is based on the basic idea that having a truly effective system means more than just putting out fires all day. That's why RCA starts with figuring out how, where, and why the issue appeared. Then it goes further: RCA strives to respond to that answer—in order to prevent it from happening again.

Download Now: ITIL 4 Best Practice e-Books

These all-new for 2020 ITIL e-books highlight important elements of ITIL 4 best practices. Quickly understand key changes and actionable concepts, written by ITIL 4 contributors.

[Free Download >](#)



[Free Download >](#)

Originating in the field of aeronautical engineering, this method is now applied in virtually every industry, but with particular focus and benefits in [software development](#). Finding the root cause of a software or infrastructure problem is a highly effective, quality engineering technique that many industries already mandate in their governance.

Root cause analysis [is considered](#) a reactive management approach. In the ITIL® framework for service management, for instance, incident management is a reactive move where you're responding to a critical incident. Problem management, on the other hand, is a proactive approach wherein you're seeking out problems to address. ([Learn more in Incident Management vs Problem Management.](#))

Why is root cause analysis necessary?

RCA has a wide range of advantages ([detailed below](#)), but it is dramatically beneficial in the continuous atmosphere of software development and information technology for two main reasons:

- **RCA focuses on cause, not symptoms.** RCA pinpoints the factors that contribute to the problem or event. But its depth also helps you avoid the temptation to single-out one issue, over others, in order to resolve the problem as fast as possible. It also helps to find the actual cause of the problem as opposed to just fixing resulting symptoms.
- **RCA significantly reduces cost and time spent by catching problems early.** Identifying the problem's root in the early stages enables developers to maintain an [agile environment](#) and drive process improvement.

Even though performing root cause analysis might feel time consuming, the opportunity to eliminate or mitigate risks and root causes is undeniably worthwhile.

RCA principles

Some of the basic [principles](#) of RCA can help organizations ensure they are following the correct methodology:

- Focusing on corrective measures of root causes is more effective than simply treating the symptoms of a problem or event.

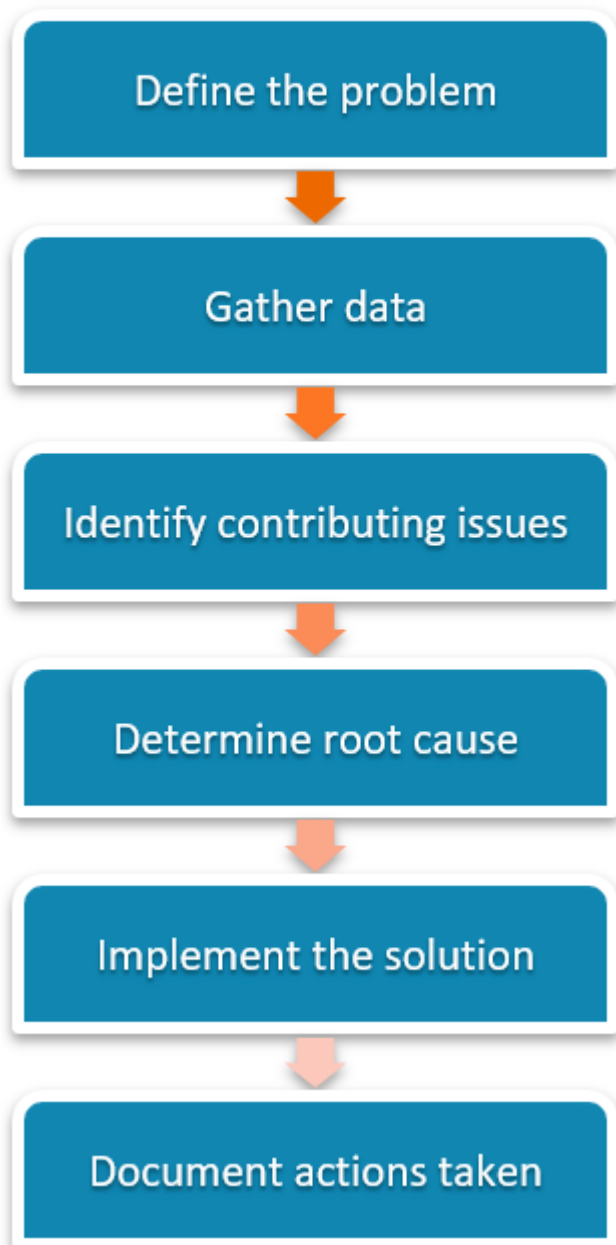
- Effective RCA is accomplished through a systematic process with evidence-backed conclusions.
- There is usually more than one root cause for a problem or event
- The focus of RCA, via problem identification, is WHY the event occurred—not who made the error.

How to perform root cause analysis

The specific map of root cause analysis may look slightly different across organizations and industries. But here are the most common steps, in order, to perform RCA:



Root Cause Analysis: 6 Steps



Let's look at these steps in detail.

1. **Define the problem.** When a problem or event arises, your first move is to contain or isolate all

suspected parts of the problem. This will help contain the problem.

2. **Gather data.** Once you find the problem, compile all data and evidence related to the specific issue to begin understanding what might be the cause.
3. **Identify any contributing issues.** You might have hands-on experience or stories from others that indicate any additional issues.
4. **Determine root cause.** Here's where your root cause analysis really occurs. You can use a variety of RCA techniques (detailed below). Each technique helps you search for small clues that may reveal the root cause, allowing the person or team to correctly identify what went wrong.
5. **Implement the solution.** Determining the root cause will likely indicate one or several solutions. You might be able to implement the solution right away. Or, the solution might require some additional work. Either way, RCA isn't done until you've implemented a solution.
6. **Document actions taken.** After you've identified and solved the root problem, document the problem and the overall resolution so that future engineers can use it as a resource.

Even if you don't expect the problem to occur again, plan as if it will.

Remember, in order to have an effective RCA it is important that the team recognizes that processes cause the problems not people. Pointing fingers and placing blame on specific workers will not solve anything.

([Learn more about the importance of a blameless culture when performing an incident postmortem](#)—the final step of your root cause analysis.)

Methods for root cause analysis

You can perform RCA using a variety of techniques. We highlight four well-known RCA techniques below—use the technique that meets your specific situation. Here's a simple distinction:

- A 5-Why analysis is good for initial troubleshooting.
- Fishikawa diagrams are helpful for identifying **all** possible root causes for a situation.
- Pareto charts help you prioritize which root causes should be addressed first, based on how often each identified root cause occurs.
- Scatter Plots are helpful in situations where you can identify and collect data on fluctuating variables that are related to the problem you are studying.

Take a look at these options and consider which might be best for your situation:

The 5-Whys

One of the simplest and most commonly utilized tools in conducting an RCA is the 5 Whys method. Mimicking curious children, the 5 Whys method literally suggests that you ask "Why?" five times in a row in order to identify the root cause of basically any process or problem.

5-Why analysis is effective because it is easy to use for solving problems where there is a single root cause.

Even though the method seems explicit enough, this approach is still meant to be flexible depending on the scenario. Sometimes five whys will be enough. Other times, you'll need to ask "Why?" a few more times. Or, you could use additional techniques to identify the root cause.

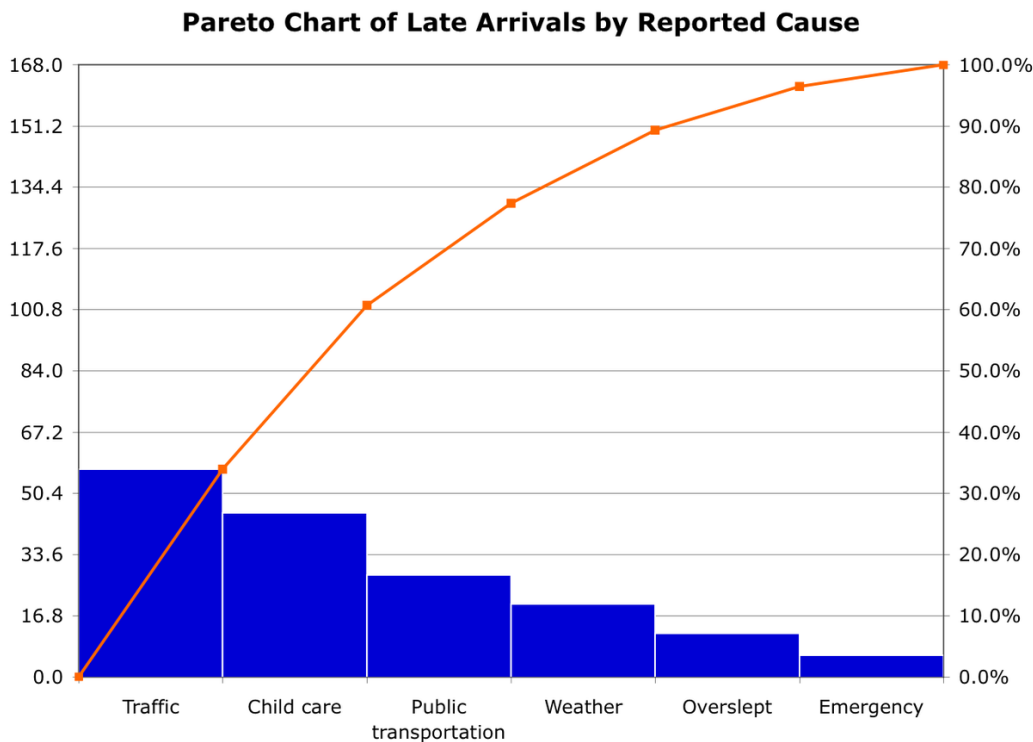
To begin this method, follow this outline:

1. Write down the specific problem that needs to be fixed, describing it completely.
2. Ask “Why?” the problem happened. Write the answer below.
3. If your first question did not find the root cause, ask “Why?” again and write that answer down.
4. Continue this process until the team agrees you’ve identified the root cause of the problem.

([See the 5 Whys in action with a simple RCA example, below.](#))

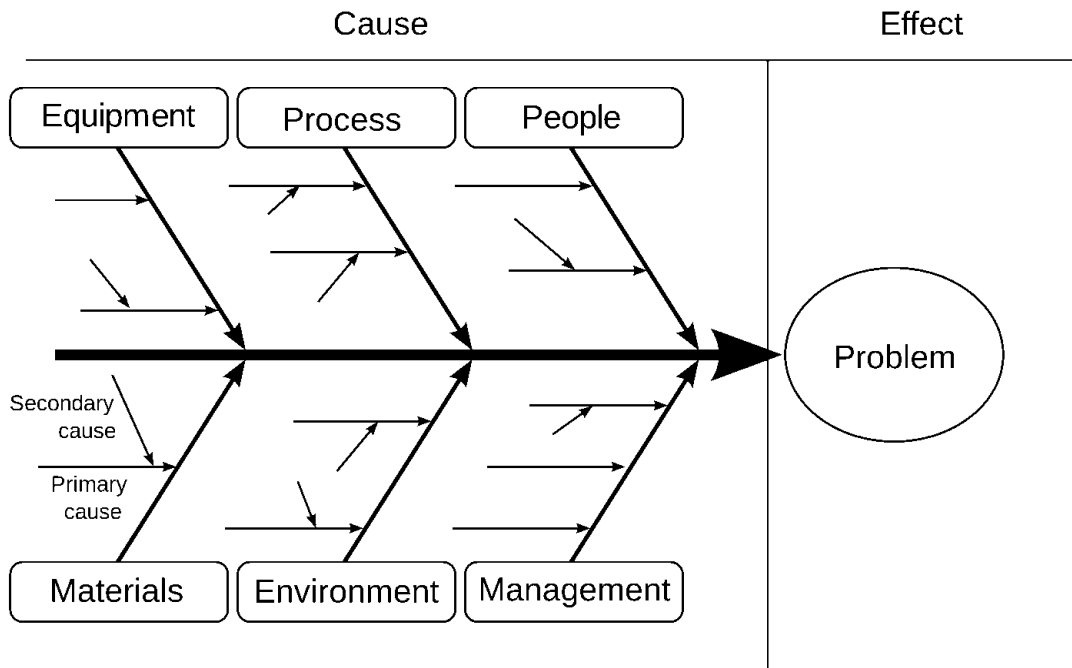
Pareto Charts

[Pareto charts](#) identify the most significant factor among a large set of factors causing a problem or event. A Pareto chart is a combined bar and line chart, where the factors are plotted as bars arranged in descending order. The chart is accompanied by a line graph showing the cumulative totals of each factor, left to right.



Ishikawa Diagrams

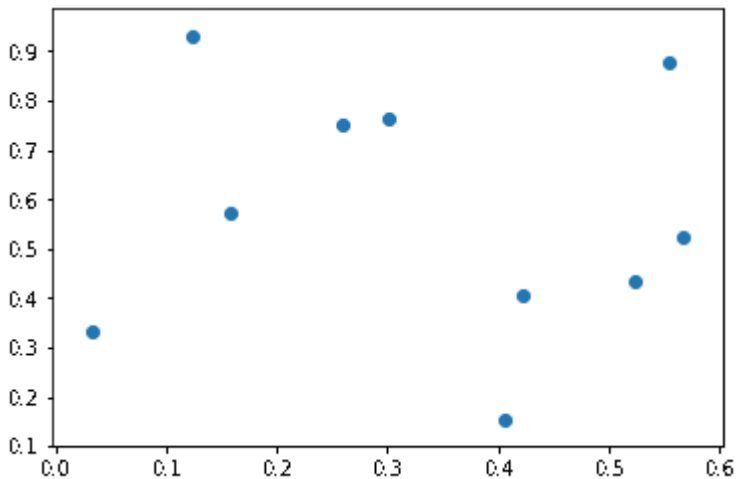
You might know the [Ishikawa Diagram](#) by other names: the fishbone, the herringbone, the cause-and-effect, and, our favorite, the Fishikawa diagram.



The Ishikawa diagram is a great visualization tool for brainstorming and discovering multiple root causes. It is shaped like a fish skeleton, with the head on the right and the possible causes shown as fishbones to the left.

Scatter Diagrams (Plots)

[Scatter diagrams](#), or scatter plots, use [regression analysis](#) to graph pairs of numerical data to determine relationships. This is helpful to identify problems and events that occur because of fluctuating measurements, such as capacity issues that happen when server traffic increases.

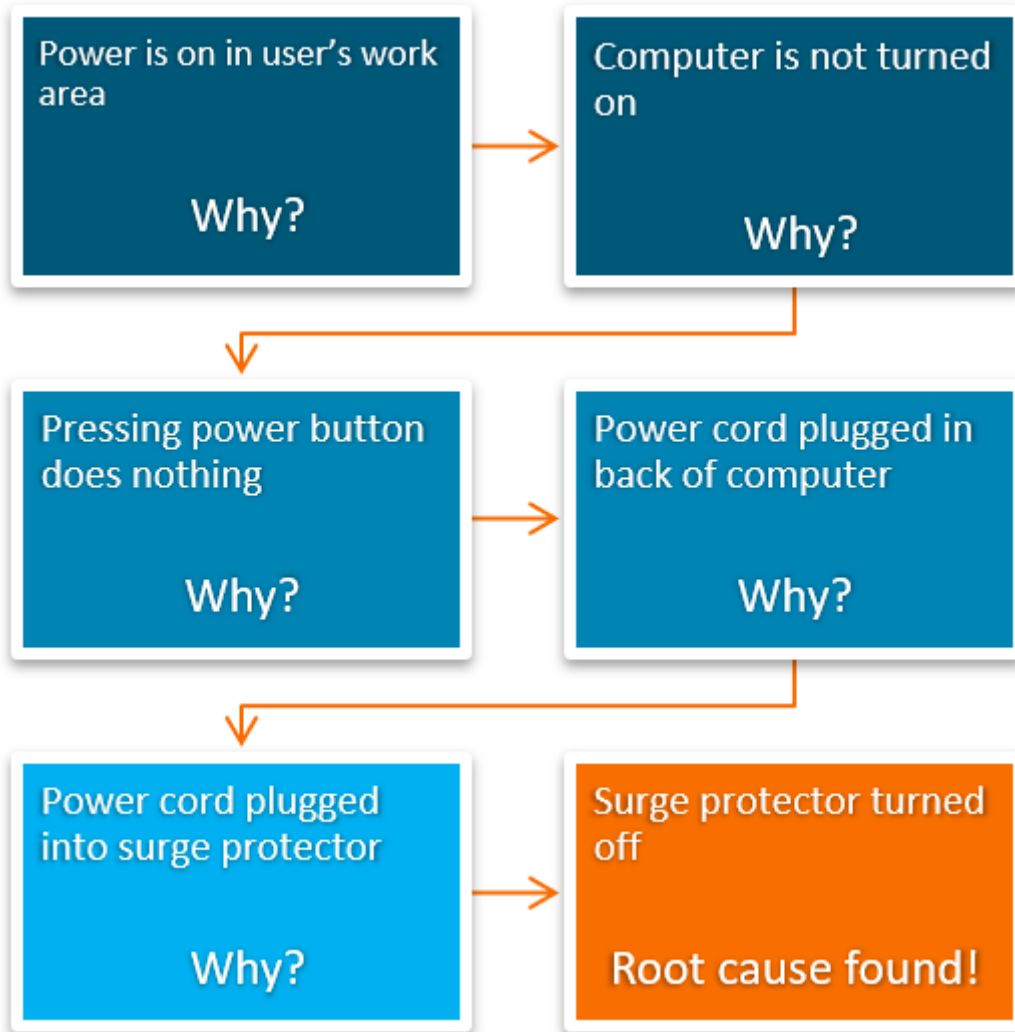


([Learn how to create your own scatter plots using Matplotlib.](#))

RCA example using 5-Why Analysis

Here is a simple 5 Why analysis where we try to determine why a computer is not turning on. At each step, we ask why the computer is not turned on. We gather data as we follow the power flow, until we finally determine that the power strip the computer plugged into is turned off.

Here's what the user has reported: Their desktop computer is not turning on. The monitor is turned on, but the user does not hear the computer fan running, and there are no power lights.



Root cause analysis using the 5 Whys to troubleshoot a computer that won't turn on

Resolution: Technician turned on the surge protector and the computer came back on again.

Benefits of root causes analysis

The main benefit of root cause analysis is obvious: identifying problems so you can solve them. RCA offers plenty more benefits that help to solidify its usefulness and importance in the tech environment.

Solve real-world problems

When the right employees get the right RCA and resolution training, you'll execute correct processes and solve common business problems.

Lower costs

When you catch problems quickly, you reduce the likelihood that those problems will turn into major incidents—especially when RCA is used to support an agile environment. RCA saves valuable employee time and ensures the organization doesn't other fines or compromises.

Make the workplace safer

Employee safety is vital, and root cause analysis provides an added peace-of-mind. By quickly and effectively investigating any safety incidents, you can solutions can be put into place to prevent anything similar from happening again down the line.

Implement effective, long-lasting solutions

When you follow RCA analysis all the way through to final documentation, you focus on long-term prevention. It also shows that your organization prioritizes solutions—not speedy workarounds.

This forward thinking enables companies to become proactive and productive.

Resolve technical debt, strengthen code base

An RCA may show the problem is [broken code due to technical debt](#). If the problem occurred due to changed business requirements, code development compromises, poor coding practices, or software entropy, the real solution may be refactoring rather than patching. Refactoring realigns your code with desired business outcomes, eliminates technical debt, and brings it up to current standards for future agile deployments.

Effective RCA saves more than money

Taking the time to create a robust root cause analysis process may take some time and effort in the initial stages, but it is an investment that will extend far beyond the expenses. The skills learned during the RCA process can be carried over to almost every other problem or field and initiate an attitude of [continuous improvement](#)—and even [innovation](#).

This culture will surely permeate your organization for the better.

Additional resources

For more on this topic, explore these resources:

- [BMC Service Management Blog](#)
- [BMC Business of IT Blog](#)
- [Data Visualization Guide](#), with tutorials on creating charts and graphs
- [How To Build Your ITSM Business Case \(Free Template Included\)](#)
- [Resilience Engineering: An Introduction](#)